https://doi.org/ 10.33472/AFJBS.6.Si2.2024.1266-1276

AFJBS

African Journal of Biological Sciences

Journal homepage: http://www.afjbs.com

Research Paper Open Access

CLOUD-POWERED SECURE LOG STORAGE USING PROOF OF STAKE AND BLOWFISH ENCRYPTION

Sharmila G[#1], Varalakshmi I[#2], Mohamed Mohsin A[#3], Ruthresh R[#4], Dharanidharan S[#5]

[#1,2]Assistant Professor

[#1,2,3,4,5]Department of Computer Science and Engineering,

[#1,2,3,4,5]Manakula Vinayagar Institute of Technology, Puducherry, India.

[1]`sharmi.deep@gmail.com`

Article History

Volume 6,Issue Si2, 2024

Received:29 Mar 2024

Accepted : 28 Apr 2024

doi: 10.33472/AFJBS.6.Si2.2024.1266-1276

Abstract:

In the paper, we can implementing innovative log storage system that makes use of smart contracts, Proof of Stake (PoS) consensus, and Blowfish encryption on a blockchain. PoS involves stakeholders in transaction validation based on their interest in the network, which improves security and energy efficiency. Logdata is protected from breaches and unwanted access with blowfish encryption. The Ethereum Blockchain Network uses Proof of Stake (PoS) to preserve network integrity and enable file retrieval requests. The Interplanetary File System (IPFS), which offers decentralized storage and censorship resistance, is used to store encrypted log data. Data integrity is confirmed using content addressing. Automating user registration while maintaining transparency and accountability is possible using a smart contract architecture. This comprehensive approach is perfect for businesses looking for cutting-edge blockchain solutions for user registration and logstorage since it puts data security, integrity, and system scalability first.

Keywords: Proof of Stake (PoS) consensus, Blowfish encryption, Smart contracts, Ethereum Blockchain Network, Interplanetary File System (IPFS).

1.0 INTRODUCTION

The necessity for effective and safe log storage solutions has grown in the current digital era. By utilizing cutting-edge technologies like the Proof of Stake (PoS) consensus mechanism, Blowfish encryption, and smart contracts, our paper seeks to address this difficulty by completely changing the way user registration and log data management are carried out. Choosing the Proof of Stake (PoS) consensus method over more conventional Proof of Work (PoW) systems is based on its higher security and energy efficiency. By using proof of stake (PoS), we make sure that only reliable users are engaged in verifying and appending new blocks to the blockchain, lowering the possibility of malicious attacks, and enhancing system speed.

Logs are an essential source of evidence for system problems, helping forensic investigators identify attack trends. Ensuring the security of a business requires constant log monitoring and verification in order to spot any weaknesses that hackers may exploit. Short-term monitoring methods guarantee real-time detection and stop protracted assaults [1]. The immutable log storage provided by blockchain technology improves cloud environment security. Our research focuses on an architecture that leverages an interplanetary system and blockchain to address file management issues. Efficient searching lowers gas consumption and improves computing performance by using smart contract mapping rather than arrays. This method successfully protects systems by optimizing log retention and threat detection.

In addition to Proof of Stake (PoS) for further security, we encrypt log data using Blowfish encryption. With predetermined criteria, smart contracts expedite user registration while improving accountability and transparency. These solutions provide businesses with strong data management, guaranteeing safe log storage and effective registration procedures. Advanced pattern recognition and speedy security patch distribution help to mitigate threats even in the face of difficulties with network monitoring and log audits. Large log file archiving creates storage issues that need for scalable solutions and continuous security device optimization in order to effectively monitor systems for cyber-attacks.

2.0. Related Work

A distributed immutable ledger known as blockchain keeps track of all transactions made by users in a network and forbids data manipulation by design. It does this by keeping an ongoing list of blocks. In 2008, Bitcoin debuted blockchain technology, laying the groundwork for its future as a cryptocurrency independent of reliable third parties. by Hitesh et al [2]. Deployed on blockchain networks, smart contracts record state changes and operate as decentralized agreements that are available to all parties by automating predetermined business logic in response to certain situations. Unlike HTTP, IPFS is a peer-to-peer distributed file system that uses content addressing to provide universal file access across computing devices. Files are assigned unique hashes for retrieval, and hashes are updated as content changes. Together, these technologies provide decentralized, automated, and secure data management and transaction execution for a range of sectors and applications.

Data integrity is guaranteed by blockchain technology's immutable distributed ledger, which is essential for examining the Lchain scheme that makes use of IPFS, Smart Contracts, and Blockchain by Parin Patel and Hiren Patel [1]. While IPFS improves decentralized storage and search performance via cryptographic hash values and content addressing, Ethereum's smart contracts enable decentralized operations without a central authority through the use of Ether and Turing programming. Through the integration of Owners, File Retrieval, Encrypted Log Files, Blockchain Upload, and Query Requests, the Lchain solution guarantees safe data management, access, and verification using Ethereum Blockchain smart contracts.

Log Generators, Logging Client or Logging Relay, Logging Cloud, and Log Monitor are the four main functional components that make up the overall architecture of the cloud-based secure log management system. Log Data is created by log generators and sent to the logging client for preparation before being sent to the logging cloud for long-term archiving by Indrajit Ray et al [9]. Organizations who subscribe can obtain storage and maintenance services through The Logging Cloud. Log monitors oversee requests for data extraction, analysis, rotation, and deletion in addition to reviewing log data. The logging cloud facilitates asynchronous communication between components via an unencrypted network. A secret-sharing scheme is used to handle key management for integrity and secrecy, along with preventative steps to lessen possible threats.

In recent years, blockchain technology has drawn a lot of interest, with several papers and applications popping up all over the world. Blockchain technology has advanced thanks to well-known papers like Ethereum, Hyperledger, InterLedger, and Steem. Major Internet corporations are participating in Hyperledger, an initiative started by the Linux Foundation that focuses on digital technologies and transaction verification. Since its launch in late 2013, Ethereum, which is well-known for its smart contract idea, has gained traction. Value movement across several blockchains inside InterLedger's architecture is facilitated by Jiansen Huang et al [4]. These initiatives offer frameworks for creating blockchain applications. Additionally, whereas ventures like IBM and Samsung's ADEPT employ blockchain to provide decentralized Internet of Things solutions, Bitmessage prioritizes user privacy in P2P communication using blockchain technology. Blockchain as a distributed database is suggested by BlockchainDB. Our suggested blockchain-based log system includes log storage, querying, and collecting, just like conventional log systems. Log data is gathered by a number of units that receive, filter, and format it into a single format before transferring it to a blockchain for analysis and storage.

3.0. Proposed Work

The study used a number of different technologies, including information gathering, datasets preparing, information dividing, constructing models, model training, validation of models, and evaluation of models.

3.1 Proof of Stake

A consensus method called Proof of Stake (PoS) uses participants' stakes rather than computing power to ensure our log storage solution. It enhances system security by discouraging bad conduct and rewarding honest engagement. PoS is the perfect solution for guaranteeing log data integrity in a blockchain-powered cloud, resulting in a scalable, secure, and lucrative structure. It also encourages economic agreement among stakeholders.

3.2 Generate Log File

Automated log creation based on system events, log entry structure definition, and storage policy implementation with naming standards and rotation techniques are all necessary to build a strong log file system. Accomplished with features like log levels, filters, timestamps, event descriptions, and severity levels are security and accessibility for authorized users. To restrict log file size, safeguard sensitive information, and stop unwanted access or modification, preservation strategies and access controls are essential. A well-designed log file system is essential for tracking performance, identifying issues, keeping an eye on system activities, and guaranteeing system security and dependability.

3.3 Extract Log Data and Detect Changes

The process of extracting log data, which is essential for troubleshooting and system monitoring, entails parsing, filtering, and transforming log data into forms that may be used, such as CSV or JSON. By enabling users to choose sources, schedule extraction, and establish parameters, automated log extraction streamlines this procedure.

Methods including event correlation, threshold-based detection, machine learning, and pattern matching are used in anomaly detection. Regular expressions are used in pattern matching to locate certain log items. In threshold-based detection, changes are detected by setting thresholds for measurements and events. Using historical data, machine learning systems find odd trends. By connecting related occurrences, event correlation can reveal patterns or sequences of events that point to problems or modifications in the system.

3.4 Encryption Process

A common encryption technique that is well-liked for its simplicity, quickness, and robust encryption powers is blowfish. Key sizes ranging from 32 to 448 bits are supported, and it runs on 64-bit blocks. Feistel network with 16 rounds of substitution and permutation, block division, key expansion using P-array and S-boxes, first bit rearrangement, round-based S-box substitution and XOR operations, and a final permutation are all part of the encryption process.

Blowfish's strong security features and effective performance make it a popular choice even with suggestions for bigger key sizes to improve resistance against brute-force assaults. It balances encryption strength with computational performance, making it appropriate for a range of encryption requirements and guaranteeing data secrecy and integrity in cryptographic applications.

3.5 Database

An IPFS cloud database's decentralized, content-addressed storage architecture is used for data management and archiving, spreading data among nodes for resilience without the need for a central server. Historical monitoring and auditability are made easier by IPFS's versioned, hashed, and immutable data. Additional precautions like encryption and access limits are required to guarantee data security and privacy. Combining IPFS with database technology allows for efficient decentralized data management that takes into account real-time performance considerations like latency and bandwidth use while providing benefits like fault tolerance, versioning, and data distribution.

3.6 Proposed Process

The log storage system is a component of a cutting-edge blockchain paper that involves a multi-step process that includes creating log files, encrypting them using the Blowfish technique, extracting data and detecting changes, and storing them in the IPFS cloud.

First, in order to create logs for monitoring, analysis, and auditing reasons, user activities, system events, failures, and timestamps are recorded. The log files are encrypted using Blowfish when they are generated, protecting their secrecy and integrity during transmission and storage. Encryption ensures that private information in the logs is protected from potential data breaches, unwanted access, and transmission problems.

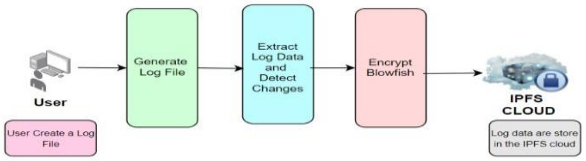

User
User Create a Log File
Generate Log File
Extract Log Data and Detect Changes
Encrypt Blowfish
IPFS CLOUD
Log data are store in the IPFS cloud

Fig 1: Process of Log secure storage.

The log file goes through an important step of being posted to the IPFS cloud after it has been encrypted using Blowfish. InterPlanetary File System, or IPFS for short, is a distributed, decentralized peer-to-peer network infrastructure that facilitates effective data sharing, retrieval, and storing amongst networked nodes. Compared to conventional centralized storage systems, the log storage system benefits from increased data resilience, decreased latency, and higher scalability thanks to the use of IPFS.

The process of extracting log data involves uploading files to the IPFS cloud and then utilizing a secure key to decrypt the files. This allows for analysis, reporting, and auditing of the original log contents to get insights into data integrity and system performance. Change detection algorithms monitor system activity and security issues to quickly identify alterations and assist in timely threat responses. After extraction, log data is safely stored in the IPFS cloud, where decentralized access restrictions, cryptographic hashing, redundancy, and storage stability ensure regulatory compliance and compliance with data regulations. This method enhances data security, resilience, and accessibility by integrating encryption, log generation, IPFS storage, and change detection techniques. It integrates state-of-the-art technology with industry best practices. It is a dependable option for enterprises giving secure log storage and management a high priority in modern computing settings because of its flexibility in responding to changing threats and capacity to preserve data integrity.

4.0. METHODOLOGY

4.1 Blowfish Encryption:

Data in cloud-based log storage systems may be securely encrypted using Bruce Schneier's symmetric-key technique, blowfish encryption. Large amounts of log data may be efficiently encrypted thanks to its variable-length block feature. Depending on their security requirements, customers may choose the encryption levels with key strengths ranging from 32 to 448 bits. Because of its speed and ease of use, Blowfish excels for real-time log encryption with minimal impact on performance, which is essential for managing log data in cloud environments. Its ability to withstand cryptographic assaults improves security even further, guaranteeing the safety of sensitive information. When combined with blockchain technology, Blowfish becomes a dependable option for secure log storage solutions by improving data integrity and auditability in distributed networks.

4.2 IPFS Cloud

Data integrity and resistance to manipulation on a decentralized network are ensured by IPFS (Interplanetary File System), which transforms data sharing and storage through content addressing and cryptographic hashes. Through the use of a distributed hash table (DHT) and peer-to-peer topology, each item of data is individually identified, maximizing bandwidth and minimizing redundancy among nodes for effective content retrieval. IPFS Cloud makes use of these characteristics to provide safe, scalable cloud storage that makes network-wide data uploading, sharing, and access easy. Due to its decentralized architecture, which improves resilience and performance with local caching, data availability is guaranteed even in the event of node failures or interruptions. Using cryptographic hashes and smart contracts, integration with blockchain technology improves security and immutability and is perfect for tamper-proof and verifiable data storage.

IPFS Cloud is a strong, dependable, and highly scalable cloud storage system that puts security, decentralization, and effective data management first.

4.3 Hash Value Retrieval

The fileID is an essential identifier in the Ethereum Blockchain ecosystem that facilitates the quick recovery of certain hash values associated with log files kept on the Interplanetary File System (IPFS). Users can conveniently access and check log files thanks to this connection, which also produces a transparent and secure system. In order to ensure data integrity and authenticity, the system makes use of Ethereum's immutable ledger. Hash values serve as cryptographic signatures that connect IPFS data to blockchain entries. This combination guarantees tamper-proof, decentralized log file protection. With assurances of secrecy and immutability, the effective extraction of hash values from Ethereum streamlines the maintenance of log files in cloud settings, providing clients with enhanced data protection.

4.4 Smart Contract

In the blockchain-powered log storage system, a smart contract that runs independently on the blockchain securely controls user registration and access permissions. It ensures that only authorized entities interact with the system by using encryption for access control and identity verification. In order to provide transparent and auditable registration, the smart contract registers and validates user credentials on the blockchain. In addition, it manages access tokens and permissions, automating processes like data validation, access updates, and transaction tracking to minimize fraud and human error. The dependability and integrity of the log storage system are strengthened by this automation, which also ensures secure and decentralized management of user registrations, access limits, and transaction validations.

4.5 Python Flask Web Application

The blockchain-powered secure log storage web application improves user experience and system functionality. Strong authentication, such as multi-factor authentication (MFA), is combined with easy-to-use log file management capabilities, such as version control, search, retrieval, and permissions. Access control lists (ACLs) and encryption are used to secure data, and smart contracts supported by blockchain integration are among the extra features like file integrity verification and audit trails. For blockchain-powered cloud settings, an extensive and secure log management platform is provided, together with real-time notifications and compliance procedures like audit logs that ensure adherence to data security regulations.

4.6 System Architecture

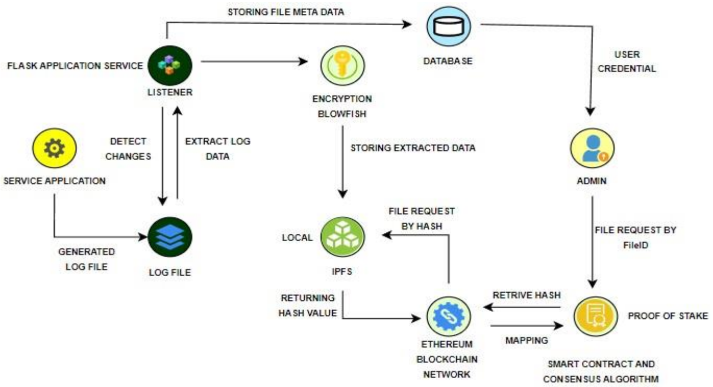


Fig 2 System Architecture of secure log storage

The log storage system is a complete solution made to handle log data effectively and securely. In order to record crucial system events from web or application servers for monitoring and auditing, a service application must first create a log file. In the background, a Flask application service keeps an eye out for modifications or new entries in the log file. It then initiates the data extraction procedure when it notices any changes, safely saving pertinent log data. Sensitive data is protected for an additional layer of security by employing the Blowfish method of encryption on the recovered log data.

The encrypted log data is then uploaded and stored on the InterPlanetary File System (IPFS), leveraging its content-addressable architecture to enhance data integrity and censorship resistance. Timestamps and IDs associated with the log file are managed separately in a dedicated database, ensuring data accessibility and integrity throughout the storage and retrieval processes, facilitating efficient organization and retrieval of log data when needed.

5.0. PERFORMANCE EVALUATION

Analysis of performance is essential for determining the efficacy and efficiency of This analysis covers a number of important topics. To determine the system's capacity to handle log storage transactions per unit of time, it is first necessary to measure the throughput and transaction speed of the system. In order to enhance procedures for real-time data processing, latency analysis is also carried out to evaluate the lag between log data production and safe cloud storage via blockchain. In order to guarantee effective use of computer resources such as CPU, memory, and storage during log storage activities, resource utilization is assessed. Scalability testing measures the system's capacity to manage growing workloads and user concurrency by conducting both vertical and horizontal scalability tests.

Moreover, in order to locate bottlenecks and enhance performance in multi-user settings, concurrency and parallelism are examined. Tests for fault tolerance and recovery guarantee system resilience and error recovery procedures, while response time measures optimize system responsiveness to improve user experience. Compiling performance limits and system competitiveness requires benchmarking against industry standards and load testing under high workloads. In order to preserve a balance between security and operational efficiency, security performance analysis evaluates the effect of security measures on system performance.

- **Execution Time:**

$$\text{Execution Time}=\text{End Time}-\text{Start Time}$$

- **Exception handling Rate:**

$$\text{Exception handling Rate} = \frac{number\ of\ \text{Exception handling}}{Total\ NUmber\ of\ Execution}$$

- **CPU Utilization Efficiency:**

$$\text{CPU Utilization Efficiency}= \frac{CPU\ Time\ Used\ by\ Code}{Total\ CPU\ Time\ Available}$$

- **Memory Utilization Efficiency:**

$$\text{Memory Utilization Efficiency} = \frac{\text{Memory Used by Code}}{\text{Total Memory Available}}$$

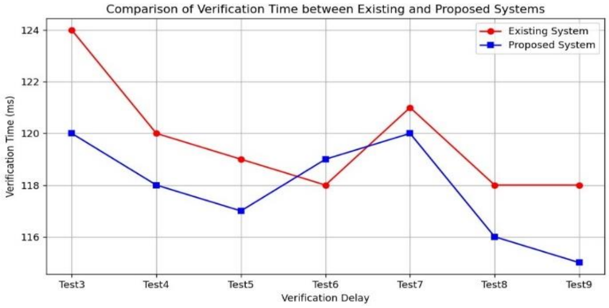


Fig.3 comparison between Proof of stake and Proof of Work of algorithm.

The number of log items varies linearly throughout the graph. The retrieval of logs is likewise faster with a larger buffer. As soon as the transaction limit was achieved, we increased the buffer size . When the client requests the log files, the validation overhead is displayed in Fig. 3. In these situations, we have performed ten comparisons to determine the validation time result. For ten log files, the suggested system model also displays a linear variance in validation time. Our suggested approach exhibits the least amount of delay in the log verification process as compared to the current one. (As shown in fig 1).

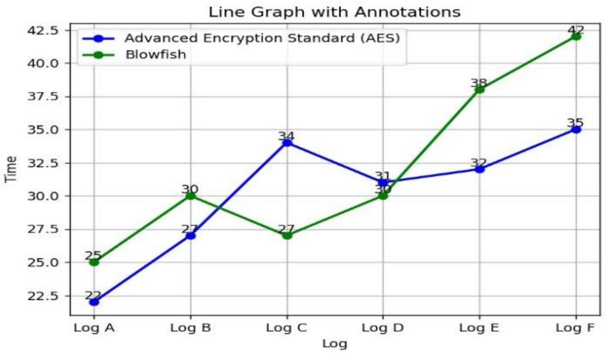


Fig 4 Comparison between AES and Blowfish encryption

In this Graph, Blowfish is well known for its quick processing speed and effectiveness, especially on systems with optimized memory and cache management thanks to its key-dependent lookup tables. This enhancement greatly increases Blowfish's performance, which makes it a preferred option in situations when quick encryption and decryption are essential.

On the other hand, AES is notable for its outstanding performance and strong security features even if it isn't always as quick as Blowfish in some situations. Its extensive use in a variety of applications is proof of its capacity to provide strong encryption with an excellent degree of data security.

AES and Blowfish's comparison study highlights the trade-off between speed and security. Because of its optimized lookup tables, Blowfish performs exceptionally quickly, whereas AES strikes a compromise between efficiency and security, making it the best option for situations where data secrecy is critical.

The decision between Blowfish and AES: Blowfish is better than AES; it increases security and makes it stronger. Ultimately comes down to the particular needs of the system, taking into account variables like processor speed, memory use, and how important data protection is.

6.0. RESULTS AND DISCUSSION

The main objective of the article is to appraise the efficacy of the present implementation by an analysis of how effectively log storage is secured through blockchain integration, utilizing the benefits of decentralized data management, immutability, and resistance to tampering. Data retrieval, encryption methods, and system performance under various loads and conditions are all examined in this assessment. In order to combat emerging risks from quantum computing, future improvements could involve implementing sophisticated encryption techniques like post-quantum cryptography. Using machine learning techniques for anomaly identification to enhance threat detection fortifies security against breaches. In order to maintain speed and reliability in the face of growing data quantities and user expectations, sustainable performance depends on streamlining database design, utilizing distributed computing frameworks for analytics and parallel processing, and putting in place caching mechanisms for quicker data retrieval.

To Maintaining regulatory compliance is essential for preserving customer confidence and legal standing, particularly in light of data protection legislation like the CCPA and GDPR. To ensure a trustworthy, secure, and future-proof system, future efforts should concentrate on quantum-resistant encryption, collaborating with experts for creative innovations, and keeping up with blockchain, cloud, and cybersecurity improvements.

The flask Python framework, which we used to build our web application, enabled us to link the webpages within it.

A. Login Page

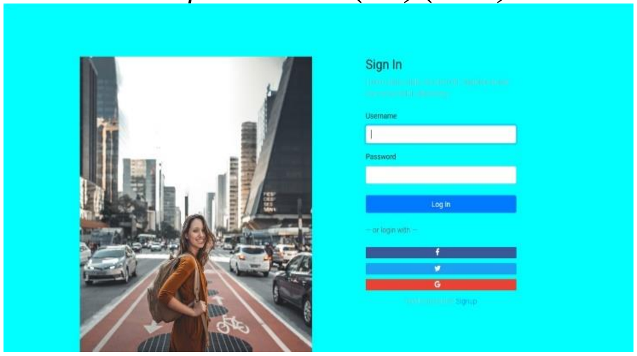
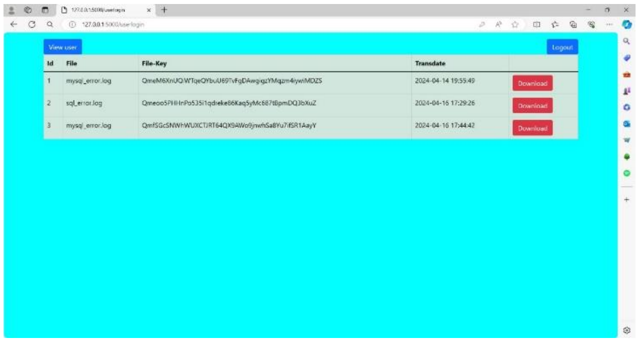


Fig 5. Home Page

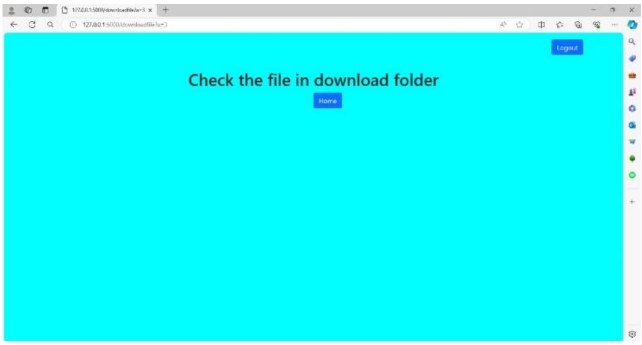
B. Log Storage Detection Page



Id	File	File-Key	Transdate	
1	mysql_error.log	QmeM6XnUQWTqeQYbuU69TvFgDAwgigzYMqzm4iywiMDZS	2024-04-14 19:55:49	Download
2	sql_error.log	Qmeoo5PHHrPo535i1qdreke86Kaq5yMc687t8pmDQ3bXuZ	2024-04-16 17:29:26	Download
3	mysql_error.log	QmfSGcSNWhWUXCTJRT64QX9AWo9jnwhSa8Yu7ifSR1AayY	2024-04-16 17:44:42	Download

Fig 6. Log Files Detection Page

C. Log storage Verification Result



Check the file in download folder

Fig 7. Verification File page

7.0 . CONCLUSION

Using blockchain technology for log storage, the article discusses noteworthy accomplishments, lessons gained, and possible future paths. The immutability, tamper-resistance, and decentralized data management of blockchain, especially when combined with IPFS and smart contracts, have significantly improved security. Important elements have been well investigated and put into practice, including blockchain-based access control, effective data retrieval methods, and encryption with algorithms like AES and Blowfish.

The paper's accomplishments in satisfying security needs and foreseeing future difficulties with data integrity and protection are demonstrated by the system's scalability, performance gains, and optimal log management procedures. Continual research and development are necessary to keep up with emerging risks and technical developments. Further research on encryption resistant to quantum assaults, the use of sophisticated machine learning techniques for anomaly detection, and speed and scalability optimization are possible future approaches.

To keep the system robust and effective, it is essential to collaborate with industry experts, check regulatory compliance continuously, and remain up to date on cybersecurity developments. In conclusion, the study shows that employing blockchain technology for safe log storage in cloud environments is both feasible and useful. The study provides a strong foundation for durable log storage solutions in the digital age with its novel methodology and answers to security problems.

REFERENCES

[1] Patel, P.., & Patel, H.. (2023). LCHAIN: A Secure Log Storage Mechanism using IPFS and Blockchain Technology. International Journal on Recent and Innovation Trends in Computing and Communication, 11(5s), 22–27. https://doi.org/10.17762/ijritcc.v11i5s.6592.

[2] Hitesh K, Shivaraja B R, Koushik N S, Puneeth K, 2023, Log Storage System – Block Chain, International Journal Of Engineering Research & Technology (Ijert) Volume 12, Issue 06 (June 2023),

[3] Z. Chen et al., "Secure Public Audit for Operation Behavior Logs in Shared Cloud Storage," 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, China, 2017, pp. 51-57, doi: 10.1109/CSE-EUC.2017.195. keywords:{Handheld computers;Conferences;Cloud computing;Scientific computing ; Ubiquitouscomputing;Privacy;Security;Public auditing;Shared Cloud Storage;Operation behavior logs;Privacy-preserving;Secure logging},

[4] J.Huang, H. Li and J. Zhang, "Blockchain Based Log System," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 3033-3038, doi: 10.1109/BigData.2018.8622204. keywords:{Blockchain;Protocols;Peer-to-peer computing Consensus algorithm; Security ; Synchronization;Reliability;Log system;the blockchain;block synchronization;nodes discovery}

[5] M. Kumar, A. K. Singh and T. V. Suresh Kumar, "Secure Log Storage Using Blockchain and Cloud Infrastructure," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bengaluru, India, 2018, pp. 1-4, doi: 10.1109/ICCCNT.2018.8494085.keywords:{Servers;Monitoring;Cloudcomputing;Securestorage;Scalability;Blockchain;Secure Log;Cloud Computing},

[6] Sucharya Deshmukh , Gaurav Jadhav , Bhumika Mhatre , Nachiket More, Rizwana Shaikh, 2023, Decentralized File Sharing using Blockchain Empowering Peer-to-Peer Collaboration: The Rise of Decentralized File Sharing, International Journal Of Engineering Research & Technology (IJERT) Volume 12, Issue 05 (May 2023),

[7] Azizi, Y., Azizi, M., & Elboukhari, M. (2022). Log Data Integrity Solution based on Blockchain Technology and IPFS. International Journal of Interactive Mobile Technologies (iJIM), 16(15), pp. 4–15. https://doi.org/10.3991/ijim.v16i15.31713

[8] Ray, K. Belyaev, M. Strizhov, D. Mulamba and M. Rajaram, "SecureLogging as a Service—Delegating Log Management to the Cloud,"in IEEE Systems Journal, vol. 7, no. 2, pp. 323- 334, June 2013.

[9] W. Huang, "A Blockchain-Based Framework for Secure Log Storage," 2019 IEEE 2nd International Conference on Computer and Communication Engineering Technology (CCET), 2019, pp. 96-100,doi: 10.1109/CCET48361.2019.8989093.

[10] H. Wang, D. Yang, N. Duan, Y. Guo and L. Zhang, "Medusa: Blockchain Powered Log Storage System," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), 2018, pp. 518-521, doi: 10.1109/ICSESS.2018.8663935

[11] Jiaxing Li, Jigang Wu, Long Chen, Block-Secure: Blockchain Based Scheme for Secure P2P Cloud Storage, Information Sciences (2018), doi: 10.1016/j.ins.2018.06.071

[12] W. Pourmajidi and A. Miranskyy, “Logchain: BlockchainAssisted Log Storage,” 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), 2018.

[13] Xu, G., Yun, F., Xu, S. et al. A blockchain-based log storage model with efficient

query. Soft Comput 27, 13779–13787 (2023). https://doi.org/10.1007/s00500-023-08975-

[14] J. Buric and D. Delija, "Challenges in network forensics," 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, 2015, pp.1382-1386.